Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления»
Кафедра «Автоматизированные системы обработки информации и управления»

Отчет

Лабораторная работа № 1

По курсу «Технологии машинного обучения»

ИСПОЛНИТЕЛЬ:

Горбатенко И.А.

Группа ИУ5-64

_____

"__"_____2020 г.

ПРЕПОДАВАТЕЛЬ:

Гапанюк Ю.Е.

_____

"__"_____2020 г.

Москва 2020

Лабораторная работа №1 по курсу "Технологии машинного обучения"

Горбатнко И.А. ИУ5-64

Цель лабораторной работы: изучение различных методов визуализация данных.

Задание:

**Выбрать набор данных (датасет). Вы можете найти список свободно распространяемых датасетов здесь. Для первой лабораторной работы рекомендуется использовать датасет без пропусков в данных, например из Scikit-learn. Пример преобразования датасетов Scikit-learn в Pandas Dataframe можно посмотреть здесь.

Выполнение:

1) Текстовое описание набора данных

В качестве набора данных возьмем базу данных наблюдаемых пациентов с возможным сердечно-сосудистым заболеванием. База состоит из 14 атрибутов:

- age - возраст пациента
- gender - пол пациента (0 или 1)
- chest_pain_type - тип боли в груди (значения от 0 до 3)
- blood_pressure - кровяное давление в состоянии покоя в мм.рт.ст.
- cholestoral - количество холестерина в мг/дл
- sugar - количество сахара в крови (1, если >120мг/дл, 0, если <=120мг/дл)
- ECG - электрокардиографические результаты в состоянии покоя (значения от 0 до 2)
- max_heart_rate - максимальное зафиксированное значение пульса
- stenocardia - наличие стенокардии или ее отсутствие после физической нагрузки (0 или 1)
- ST_depression - депрессия ST, вызванная физической нагрузкой относительно покоя
- slope - наклон пикового значения ST при нагрузке (от 0 до 2)
- vessels - количество крупных сосудов, показанных на флюороскопии (от 0 до 3)
- thal - 3 = нормальный; 6 = исправленный дефект; 7 = обратимый дефект
- target - наличие или отсутствие сердечно-сосудистого заболевания у пациента (1 или 0)

Конечной целью (target) является значение 0 или 1 (соответственно отсутствие сердечно-сосудистого заболевания или его наличие). Будем решать задачу классификации и задачу регрессии. В качестве целевого признака для решения задачи классификации будем использовать "target". "target" принимает значения только 0 или 1, значит это задача бинарной классификации. В качестве целевого признака для решения задачи регресии будем использовать "max_heart_rate". Датасет состоит из одного файла "Heart_Desease.csv", содержащий 303 строки.

Импортируем библиотеки с помощью команды:

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

Загрузим данные датасета:

In [4]:
```
data = pd.read_csv('Heart_Desease.csv', sep=",")
```

Проверим корректность загрузки:

In [5]:
```
data.head()
```

Out[5]:

	age	gender	chest_pain_type	blood_pressure	cholestoral	sugar	ECG	max_heart_rate	stenoca
0	61	1	0	148	203	0	1	161	
1	54	1	2	125	273	0	0	152	
2	71	0	2	110	265	1	0	130	
3	54	1	0	110	239	0	1	126	
4	66	1	0	112	212	0	0	132	

Уточним размер датасета:

In [6]:
```
data.shape
```

Out[6]:
```
(303, 14)
```

Список атрибутов:

In [7]:
```
data.dtypes
```

Out[7]:
```
age                 int64
gender              int64
chest_pain_type     int64
blood_pressure      int64
cholestoral         int64
sugar               int64
ECG                 int64
max_heart_rate      int64
stenocardia         int64
ST_depression     float64
slope               int64
vessels             int64
thal                int64
target              int64
dtype: object
```

Проверка датасета на наличие пустых значений:

In [8]:
```
for col in data.columns:
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
age - 0
gender - 0
chest_pain_type - 0
blood_pressure - 0
cholestoral - 0
sugar - 0
ECG - 0
max_heart_rate - 0
stenocardia - 0
ST_depression - 0
slope - 0
vessels - 0
thal - 0
target - 0
```

Основные статистические характеристки набора данных:

In [9]:
```
data.describe()
```

Out[9]:

	age	gender	chest_pain_type	blood_pressure	cholestoral	sugar	ECG
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000
mean	54.366337	0.683168	0.966997	131.623762	246.264026	0.148515	0.528053
std	9.082101	0.466011	1.032052	17.538143	51.830751	0.356198	0.525860
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000
25%	47.500000	0.000000	0.000000	120.000000	211.000000	0.000000	0.000000
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	274.500000	0.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000

Проверим уникальные значения для целевого признака:

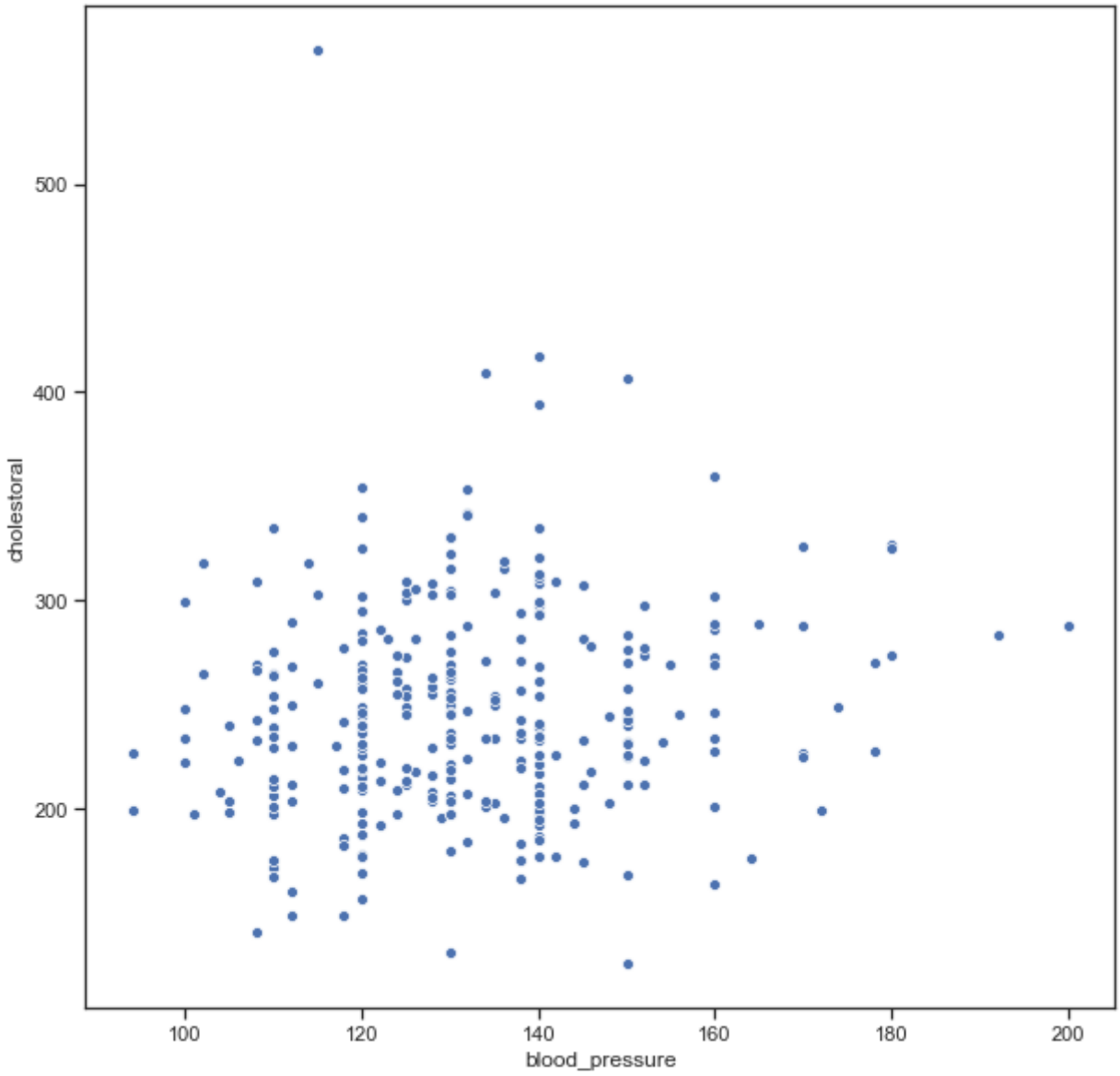
In [11]:
```
data['target'].unique()
```

Out[11]:
```
array([0, 1])
```

2) Визуальное исследование датасета

In [15]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='blood_pressure', y='cholestoral', data=data)
```

Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x1a1d524b10>

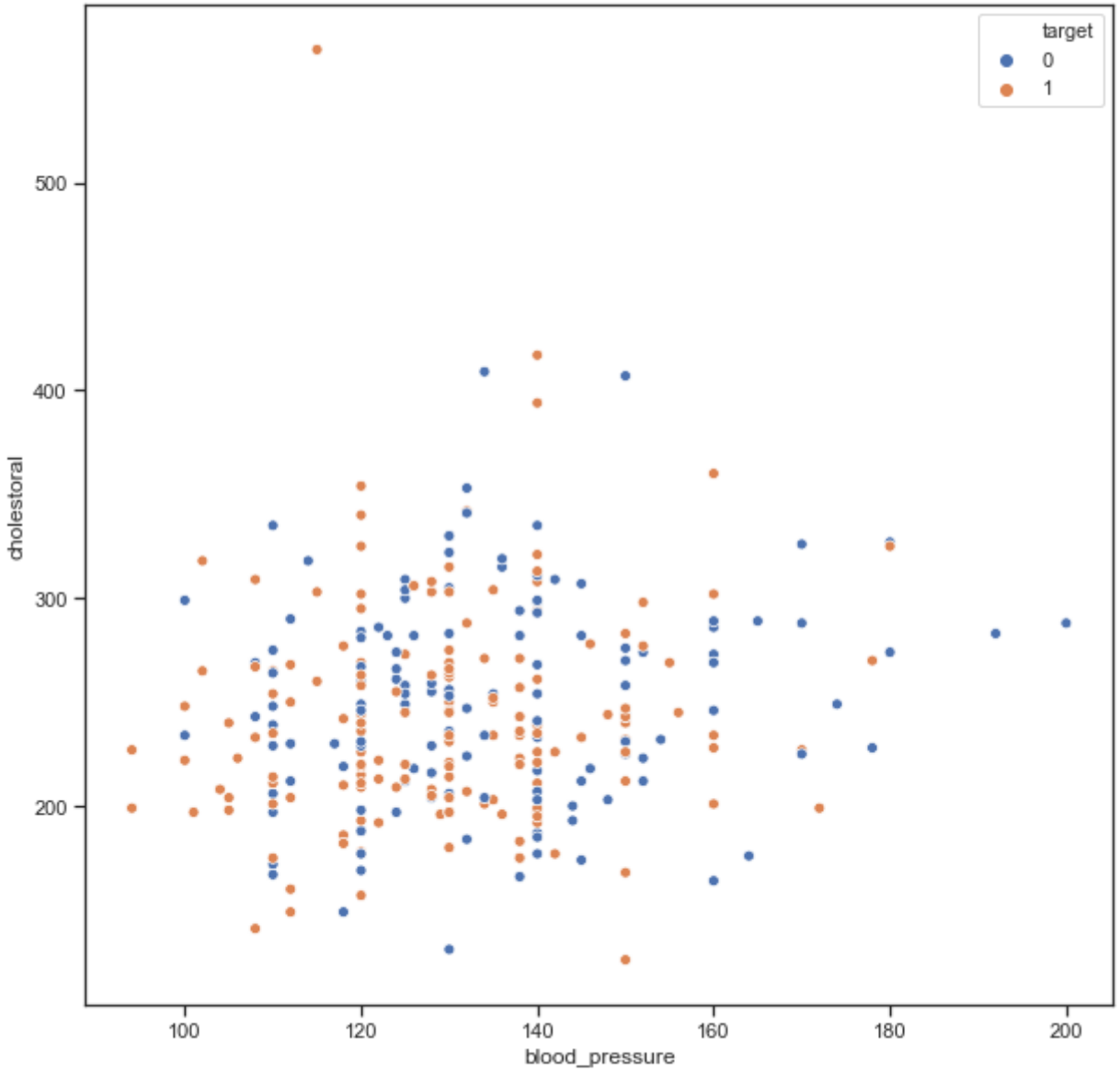


Можно видеть что между атрибутами blood_pressure и cholestoral пристутствует какая-то связь, очень и очень отдаленно напоминающая линейную зависимость.

Введем в эту зависимость целевой признак:

In [16]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='blood_pressure', y='cholestoral', data=data, hue=
```

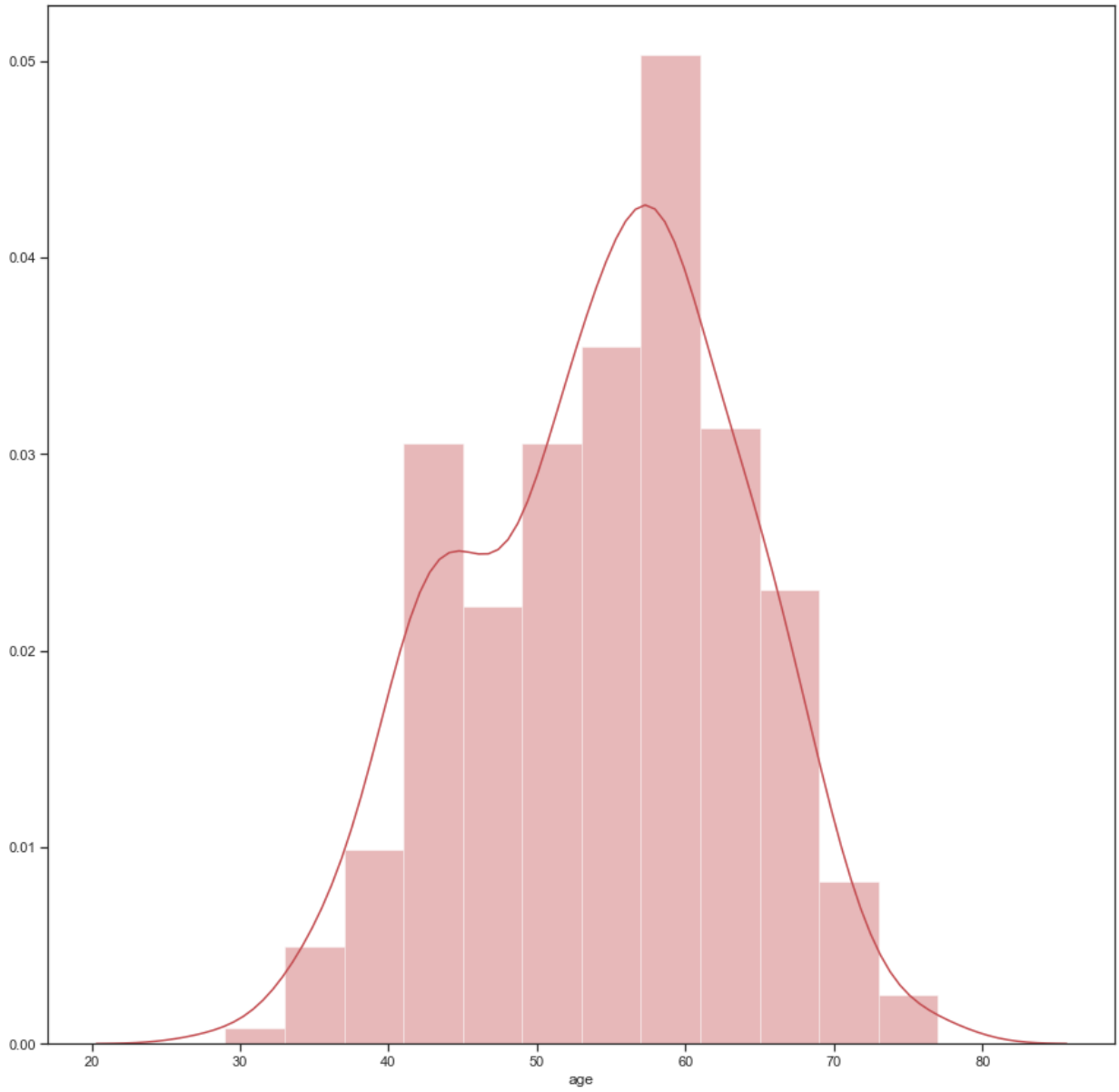
Out[16]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a1d6e61d0>
```



Можем наглядно оценить, например, распределение возрастов пациентов:

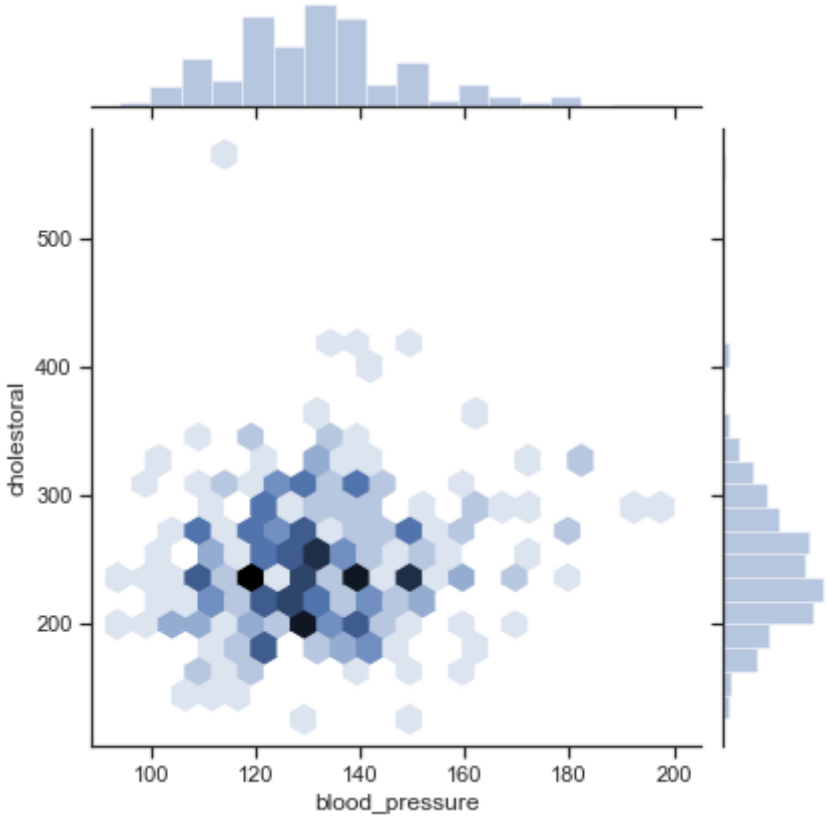
In [20]:
```
fig, ax = plt.subplots(figsize=(15,15))
sns.distplot(data['age'], color="r")
```

Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x1a1e4aae90>



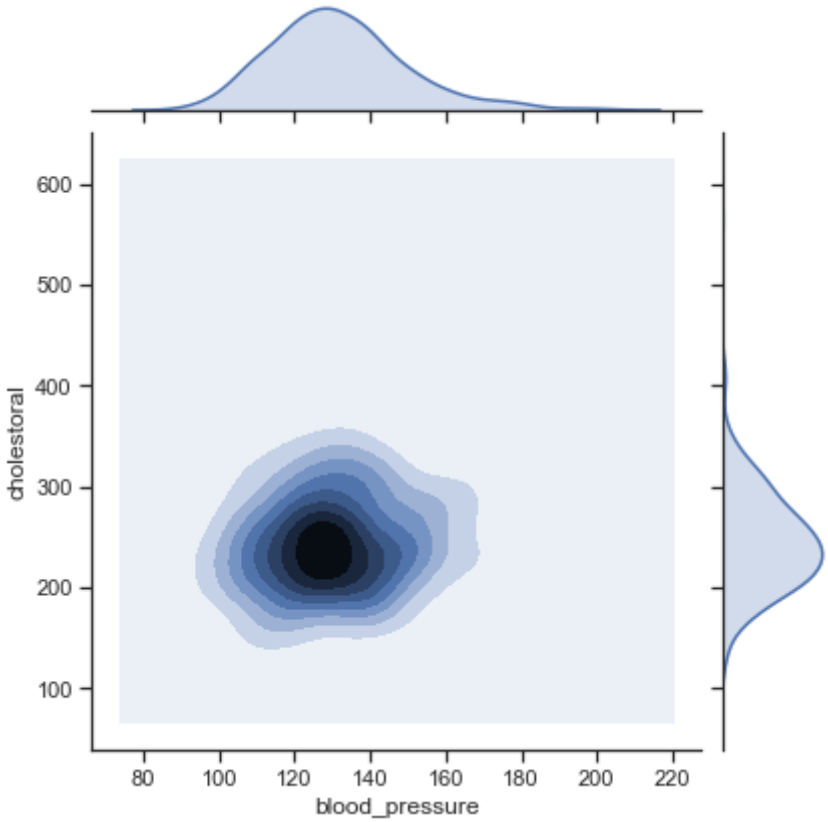
In [22]:
```
sns.jointplot(x='blood_pressure', y='cholestoral', data=data, kind="hex")
```

Out[22]:
```
<seaborn.axisgrid.JointGrid at 0x1a1e8f4450>
```



In [23]:
```
sns.jointplot(x='blood_pressure', y='cholestoral', data=data, kind="kde")
```

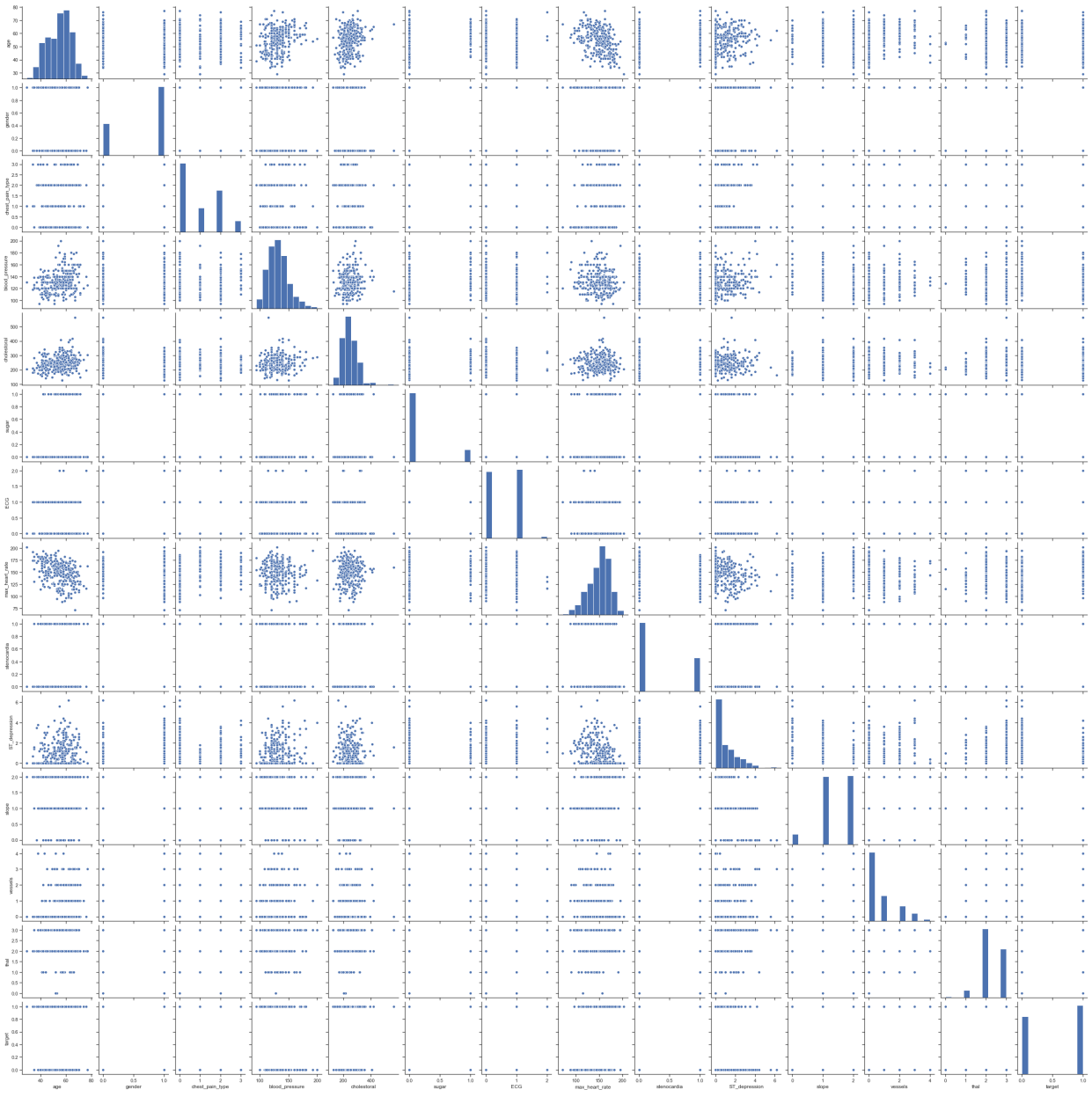
Out[23]:
```
<seaborn.axisgrid.JointGrid at 0x1a1eb0abd0>
```



Парные диаграммы:

In [24]:
```
sns.pairplot(data)
```

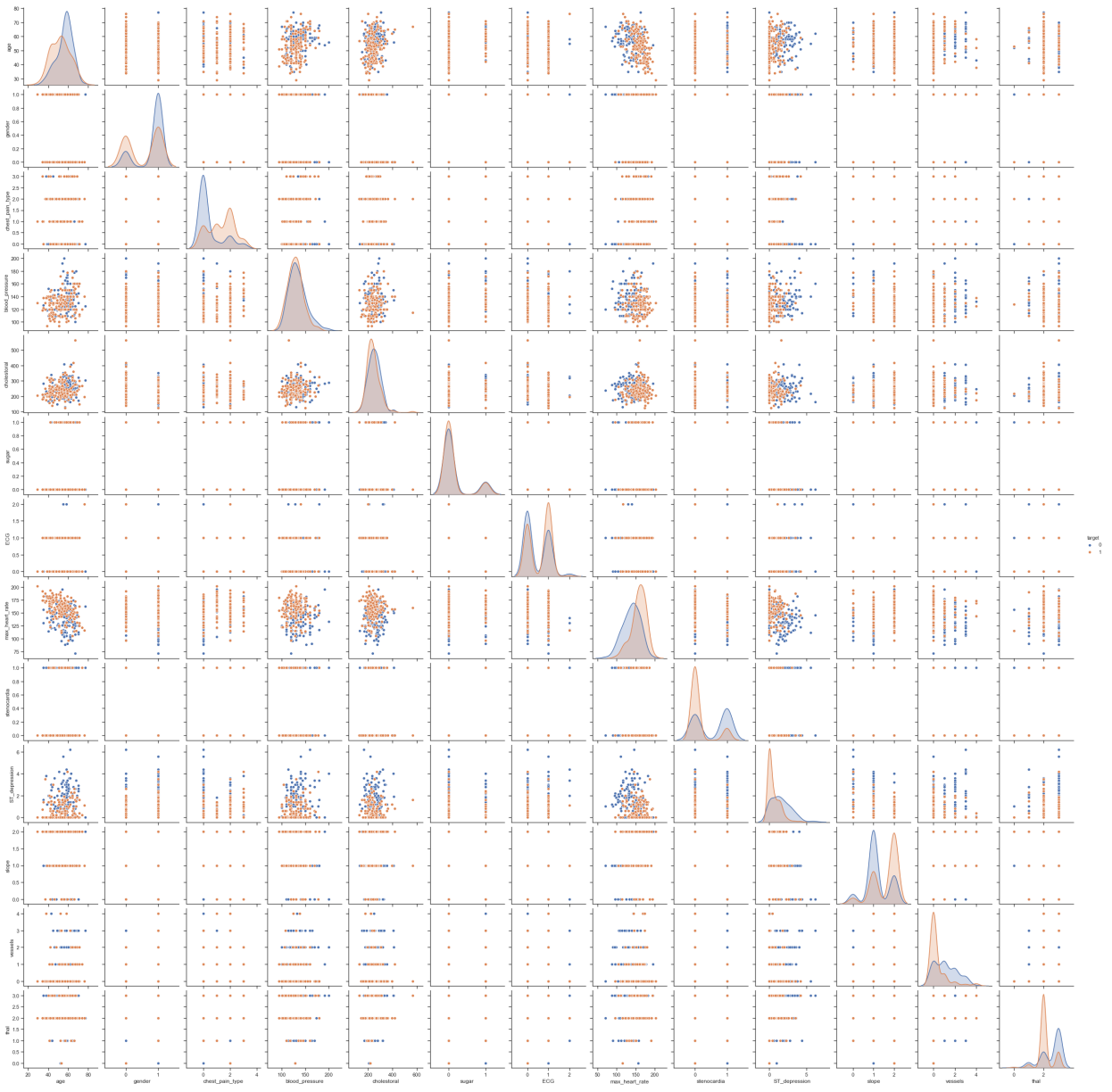
Out[24]:
```
<seaborn.axisgrid.PairGrid at 0x1a1edfac10>
```



Сгруппируем по значению целевого признака:

In [25]:
```
sns.pairplot(data, hue="target")
```

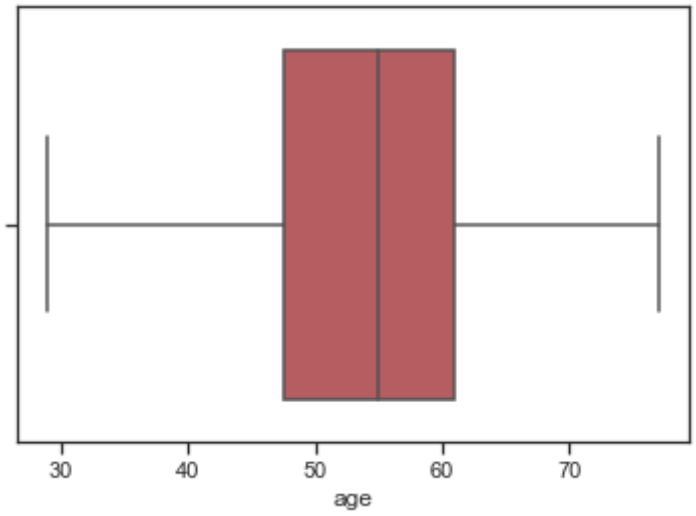
Out[25]:
```
<seaborn.axisgrid.PairGrid at 0x1a24eacb50>
```



Отобразим одномерное распределение вероятности:

In [27]:
```
sns.boxplot(x=data['age'], color="r")
```

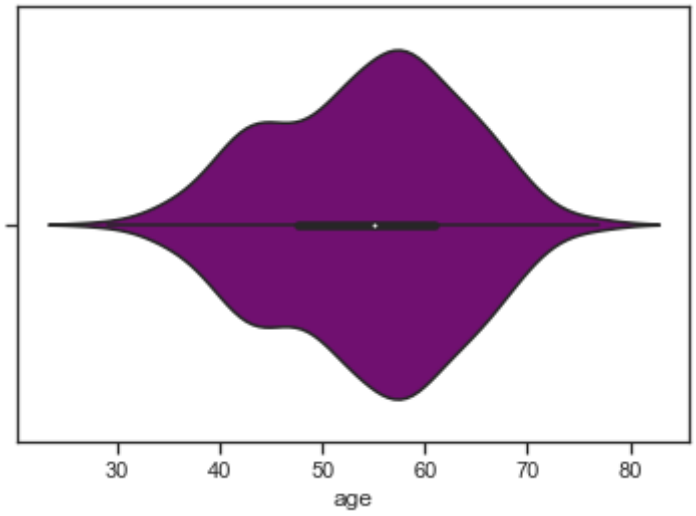
Out[27]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a2d58fd90>
```



Скрипичные диаграммы:

In [32]:
```
sns.violinplot(x=data['age'], color="purple")
```

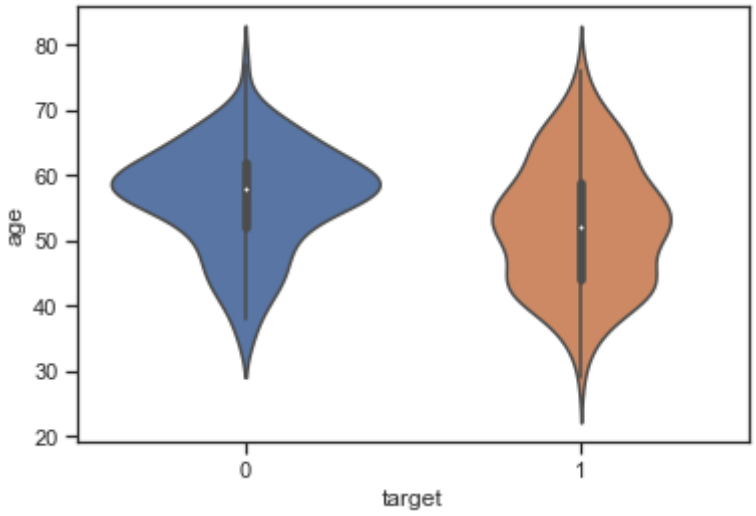
Out[32]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a2f3fec90>
```



Сгруппируем по целевому признаку:

In [36]:
```
# Распределение параметра Humidity сгруппированные по Occupancy.
sns.violinplot(x='target', y='age', data=data)
```

Out[36]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a2f77f0d0>
```



3) Информация о корреляции признаков

In [37]:
```
data.corr()
```

Out[37]:

	age	gender	chest_pain_type	blood_pressure	cholestoral	sugar	
age	1.000000	-0.098447	-0.068653	0.279351	0.213678	0.121308	-0.11
gender	-0.098447	1.000000	-0.049353	-0.056769	-0.197912	0.045032	-0.05
chest_pain_type	-0.068653	-0.049353	1.000000	0.047608	-0.076904	0.094444	0.04
blood_pressure	0.279351	-0.056769	0.047608	1.000000	0.123174	0.177531	-0.11
cholestoral	0.213678	-0.197912	-0.076904	0.123174	1.000000	0.013294	-0.15
sugar	0.121308	0.045032	0.094444	0.177531	0.013294	1.000000	-0.08
ECG	-0.116211	-0.058196	0.044421	-0.114103	-0.151040	-0.084189	1.00
max_heart_rate	-0.398522	-0.044020	0.295762	-0.046698	-0.009940	-0.008567	0.04
stenocardia	0.096801	0.141664	-0.394280	0.067616	0.067023	0.025665	-0.07
ST_depression	0.210013	0.096093	-0.149230	0.193216	0.053952	0.005747	-0.05
slope	-0.168814	-0.030711	0.119717	-0.121475	-0.004038	-0.059894	0.09
vessels	0.276326	0.118261	-0.181053	0.101389	0.070511	0.137979	-0.07
thal	0.068001	0.210041	-0.161736	0.062210	0.098803	-0.032019	-0.01
target	-0.225439	-0.280937	0.433798	-0.144931	-0.085239	-0.028046	0.13

In [38]:
```
data.corr(method='pearson')
```

Out[38]:

	age	gender	chest_pain_type	blood_pressure	cholestoral	sugar	
age	1.000000	-0.098447	-0.068653	0.279351	0.213678	0.121308	-0.11
gender	-0.098447	1.000000	-0.049353	-0.056769	-0.197912	0.045032	-0.05
chest_pain_type	-0.068653	-0.049353	1.000000	0.047608	-0.076904	0.094444	0.04
blood_pressure	0.279351	-0.056769	0.047608	1.000000	0.123174	0.177531	-0.11
cholestoral	0.213678	-0.197912	-0.076904	0.123174	1.000000	0.013294	-0.15
sugar	0.121308	0.045032	0.094444	0.177531	0.013294	1.000000	-0.08
ECG	-0.116211	-0.058196	0.044421	-0.114103	-0.151040	-0.084189	1.00
max_heart_rate	-0.398522	-0.044020	0.295762	-0.046698	-0.009940	-0.008567	0.04
stenocardia	0.096801	0.141664	-0.394280	0.067616	0.067023	0.025665	-0.07
ST_depression	0.210013	0.096093	-0.149230	0.193216	0.053952	0.005747	-0.05
slope	-0.168814	-0.030711	0.119717	-0.121475	-0.004038	-0.059894	0.09
vessels	0.276326	0.118261	-0.181053	0.101389	0.070511	0.137979	-0.07
thal	0.068001	0.210041	-0.161736	0.062210	0.098803	-0.032019	-0.01
target	-0.225439	-0.280937	0.433798	-0.144931	-0.085239	-0.028046	0.13

In [39]:
```
data.corr(method='kendall')
```

Out[39]:

	age	gender	chest_pain_type	blood_pressure	cholestoral	sugar	
age	1.000000	-0.082272	-0.071577	0.201071	0.135062	0.094595	-0.10
gender	-0.082272	1.000000	-0.057955	-0.044438	-0.124104	0.045032	-0.04
chest_pain_type	-0.071577	-0.057955	1.000000	0.027548	-0.069899	0.083862	0.06
blood_pressure	0.201071	-0.044438	0.027548	1.000000	0.086474	0.127574	-0.10
cholestoral	0.135062	-0.124104	-0.069899	0.086474	1.000000	0.015140	-0.13
sugar	0.094595	0.045032	0.083862	0.127574	0.015140	1.000000	-0.08
ECG	-0.109349	-0.048085	0.060839	-0.105147	-0.132664	-0.080996	1.00
max_heart_rate	-0.280009	-0.032817	0.246160	-0.027760	-0.031437	-0.011749	0.07
stenocardia	0.074427	0.141664	-0.390708	0.044419	0.075044	0.025665	-0.07
ST_depression	0.193269	0.086437	-0.125081	0.109103	0.035176	0.024342	-0.06
slope	-0.147713	-0.024333	0.145796	-0.070360	-0.010039	-0.044546	0.11
vessels	0.273255	0.112199	-0.189400	0.070387	0.088549	0.126434	-0.09
thal	0.070722	0.244164	-0.188999	0.049028	0.066255	-0.006559	-0.01
target	-0.197857	-0.280937	0.430506	-0.102064	-0.099131	-0.028046	0.14

In [40]:
```
data.corr(method='spearman')
```

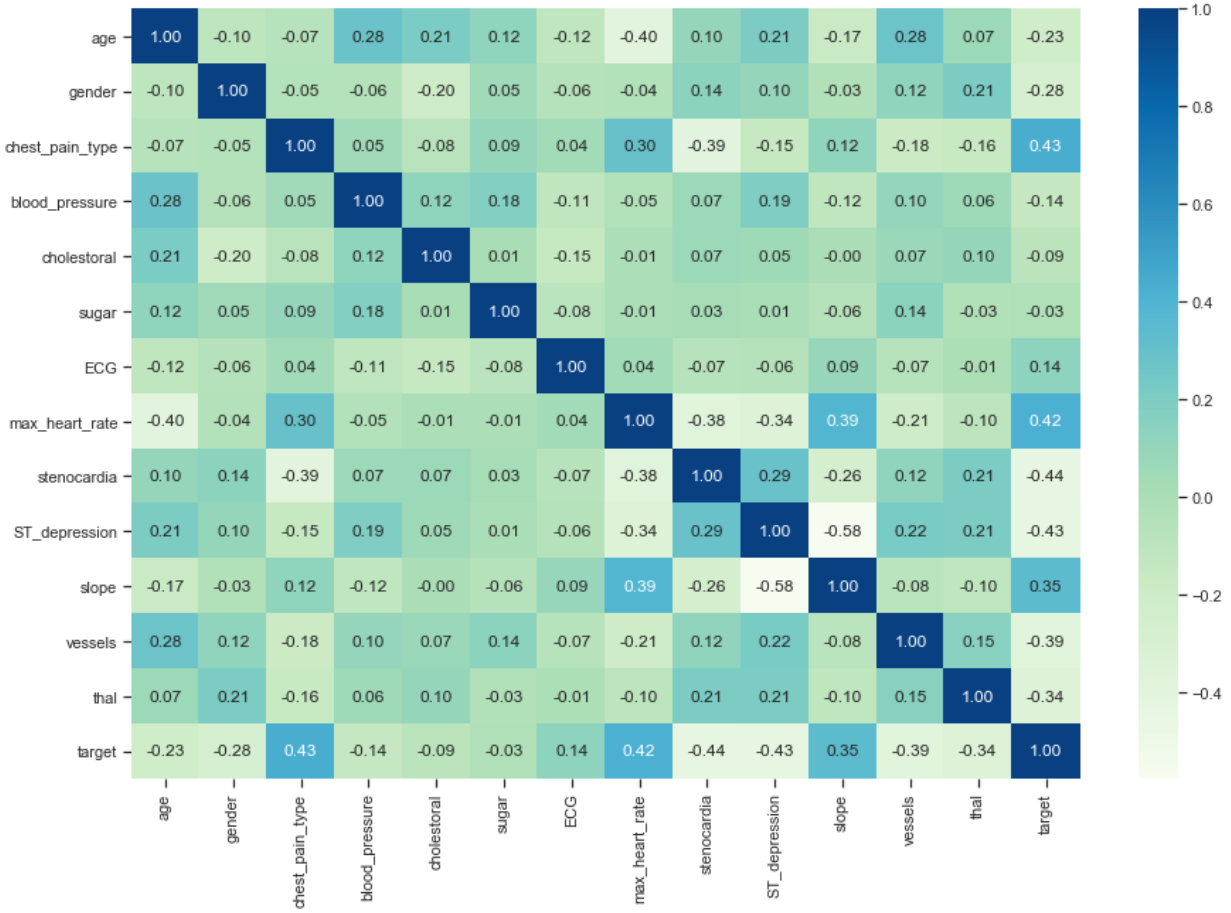
Out[40]:

	age	gender	chest_pain_type	blood_pressure	cholestoral	sugar	
age	1.000000	-0.099131	-0.087494	0.285617	0.195786	0.113978	-0.13
gender	-0.099131	1.000000	-0.062041	-0.052941	-0.151342	0.045032	-0.04
chest_pain_type	-0.087494	-0.062041	1.000000	0.035413	-0.091721	0.089775	0.06
blood_pressure	0.285617	-0.052941	0.035413	1.000000	0.126562	0.151984	-0.12
cholestoral	0.195786	-0.151342	-0.091721	0.126562	1.000000	0.018463	-0.16
sugar	0.113978	0.045032	0.089775	0.151984	0.018463	1.000000	-0.08
ECG	-0.132769	-0.048389	0.065640	-0.125841	-0.161933	-0.081508	1.00
max_heart_rate	-0.398052	-0.039868	0.324013	-0.040407	-0.046766	-0.014273	0.08
stenocardia	0.089679	0.141664	-0.418256	0.052918	0.091514	0.025665	-0.07
ST_depression	0.268291	0.100715	-0.161449	0.154267	0.045260	0.028363	-0.07
slope	-0.184048	-0.025010	0.159478	-0.086570	-0.012551	-0.045786	0.11
vessels	0.340955	0.119368	-0.216006	0.090140	0.111981	0.134513	-0.09
thal	0.087254	0.250821	-0.207840	0.059673	0.083628	-0.006737	-0.01
target	-0.238400	-0.280937	0.460860	-0.121593	-0.120888	-0.028046	0.14

Визуализируем корреляционную матрицу:

In [55]:
```
fig, ax = plt.subplots(figsize=(15,10))
sns.heatmap(data.corr(), annot=True, fmt='.2f', cmap='GnBu')
```

Out[55]: <matplotlib.axes._subplots.AxesSubplot at 0x1a31986710>



In [ ]: